

DESCRIPTION

DATA PROCESSOR AND DATA PROCESSING METHOD

TECHNICAL FIELD

5 The present invention relates to a data processor and processing method for writing stream data of a moving picture stream on a storage medium such as an optical disk.

BACKGROUND ART

10 Various types of data streams have been standardized to compress and encode video data at low bit rates. A system stream compliant with the MPEG-2 System standard (ISO/IEC 13818-1) is known as one such data stream. There are three types of system streams, namely, a program stream (PS), a
15 transport stream (TS) and a PES stream.

 Recently, another data stream, complying with the MPEG-4 system standard (ISO/IEC 14496-1), has been being defined. In a format compliant with the MPEG-4 system standard, video streams, including MPEG-2 or MPEG-4 video streams, and various
20 types of audio streams are multiplexed together, thereby

generating moving picture stream data. Furthermore, according to a format compliant with the MPEG-4 system standard, auxiliary information is defined. The auxiliary information and a moving picture stream are defined as a single file (which is called an "MP4 file"). The data structure of an MP4 file is based on, and an extension of, a QuickTime® file format of Apple Corporation. It should be noted that as for a system stream compliant with the MPEG-2 System standard, no data structure storing the auxiliary information (such as access information, special playback information and recording date) is defined. This is because the auxiliary information is included in a system stream according to the MPEG-2 System standard.

Video data and audio data would often be recorded on a magnetic tape in the past. Recently, however, optical disks such as DVD-RAMs and MOs have attracted much attention as storage media that will soon replace magnetic tapes.

FIG. 1 shows a configuration for a conventional data processor 350. The data processor 350 can read and write a data stream from/on a DVD-RAM disk. The data processor 350

receives a video data signal at a video signal input section 300 and an audio data signal at an audio signal input section 302, respectively, and sends them to an MPEG-2 compressing section 301. The MPEG-2 compressing section 301 compresses and encodes the video data and audio data in accordance with the MPEG-2 standard and/or the MPEG-4 standard, thereby generating an MP4 file. More specifically, the MPEG-2 compressing section 301 compresses and encodes the video data and audio data in accordance with the MPEG-2 Video standard to generate a video stream and an audio stream. Thereafter, the MPEG-2 compressing section 301 further multiplexes these streams together in accordance with the MPEG-4 system standard, thereby generating an MP4 file. In this case, a writing control section 341 controls the operation of a writing section 320. In accordance with an instruction given by the writing control section 341, a continuous data area detecting section 340 checks the availability of sectors being managed by a logical block management section 343, thereby detecting physically continuous unused areas. Then, the writing section 320 gets the MP4 file written on the DVD-RAM

disk 331 by a pickup 330.

FIG. 2 shows the data structure of an MP4 file 20. The MP4 file 20 includes auxiliary information 21 and a moving picture stream 22. The auxiliary information 21 is described
5 by an atom structure 23 defining the attributes of video data, audio data and so on. FIG. 3 shows a specific example of the atom structure 23. In the atom structure 23, the data size (on a frame basis), the address of the data storage location, a time stamp showing the playback timing and other pieces of
10 information are described for each of the video data and audio data. This means that the video data and audio data are managed as individual track atoms.

In the moving picture stream 22 of the MP4 file shown in FIG. 2, the video data and audio data are each arranged on a
15 frame basis, thereby making up a stream. For example, if the moving picture stream has been obtained by the compression coding method compliant with the MPEG-2 standard, then a plurality of GOPs are defined for the moving picture stream. A GOP is a unit for a collection of video frames including an
20 I-picture, which is a video frame that can be read by itself,

and P- and B-pictures that are interposed between one I-picture and the next I-picture. In reading an arbitrary video frame of the moving picture stream 22, first, a GOP including that video frame is identified in the moving picture stream
5 22.

It should be noted that a data stream with a structure including a moving picture stream and auxiliary information as in the data structure of the MP4 file shown in FIG. 2 will be referred to herein as an "MP4 stream".

10 FIG. 4 shows the data structure of a moving picture stream 22. The moving picture stream 22 includes a video track and an audio track, to each of which an identifier TrackID is added. Not every moving picture stream includes one track apiece but the tracks may sometimes be changed.
15 FIG. 5 shows a moving picture stream 22 in which the tracks are changed on the way.

FIG. 6 shows a correlation between a moving picture stream 22 and storage units (i.e., sectors) of the DVD-RAM disk 331. The writing section 320 writes the moving picture
20 stream 22 on the DVD-RAM disk in real time. More specifically,

the writing section 320 secures a logical block, which is physically continuous for at least 11 seconds when converted at the maximum write rate, as a single continuous data area and sequentially writes video and audio frames there. The continuous data area consists of a plurality of logical blocks, each of which has a size of 32 kilobytes and to each of which an error correction code is added. Each logical block is further made up of a plurality of sectors, each having a size of 2 kilobytes. The continuous data area detecting section 340 of the data processor 350 detects again the next continuous data area when the remainder of the single continuous data area becomes less than 3 seconds, for example, if converted at the maximum write rate. When the single continuous data area is full, the continuous data area detecting section 340 writes the moving picture stream on the next continuous data area. The auxiliary information 21 of the MP4 file 20 is also written on the continuous data area that has been secured in a similar manner.

FIG. 7 shows how the written data is managed by the file system of the DVD-RAM. In this case, either a file system

compliant with the universal disk format (UDF) standard or a file system compliant with ISO/IEC 13346 (Volume and File Structure of Write-Once and Rewritable Media Using Non-Sequential Recording for Information Interchange) may be used.

5 In FIG. 7, the continuously written MP4 file is stored under the file name "MOV0001.MP4". The file name and file entry location of this file are managed by a file identifier descriptor (FID). The file name is defined as MOV0001.MP4 in the file identifier, while the file entry location is defined
10 by the top sector number of the file entry in the ICB.

It should be noted that the UDF standard corresponds to the installing terms of the ISO/IEC 13346 standard. By connecting a DVD-RAM drive to a computer such as a PC by way of a 1394 interface and a serial bus protocol 2 (SBP-2), the
15 PC can also treat a file that was written in a UDF compliant format as a single file.

By using allocation descriptors, the file entry manages the continuous data areas (CDAs) a, b, c and the data area d where the data is stored. More specifically, if the writing
20 control section 341 finds a defective logical block while

writing the MP4 file on the continuous data area a, then the writing control section 341 will skip that defective logical block and continue to write the file from the beginning of the continuous data area b. Next, if the writing control section 5 341 finds a non-writable PC file storage area while writing the MP4 file on the continuous data area b, then the writing control section 341 will resume writing the file from the beginning of the continuous data area c. On having written the file, the writing control section 341 will write the 10 auxiliary information 21 on the data area d. As a result, the file VR_MOVIE.VRO is made up of the continuous data areas d, a, b and c.

As shown in FIG. 7, the beginning of the data to be referenced by the allocation descriptor a, b, c or d matches 15 with the top of its associated sector. Also, the data to be referenced by every allocation descriptor a, b or d, except the last allocation descriptor c, has a data size that is an integral number of times as large as that of one sector. Such a description rule is defined in advance.

20 In playing back an MP4 file, the data processor 350

retrieves and receives a moving picture stream by way of the pickup 330 and a reading section 321 and gets the stream decoded by an MPEG-2 decoding section 311, thereby generating a video signal and an audio signal, which are eventually
5 output through a video signal output section 310 and an audio signal output section 312, respectively. Reading the data from the DVD-RAM disk and outputting the read data to the MPEG-2 decoding section 311 are carried out concurrently. In this case, the data read rate is set higher than the data
10 output rate and is controlled such that the data to be played back does not run short. Accordingly, if the data is continuously read and output, then extra data can be obtained by the difference between the data read rate and the data output rate. By using that extra data as the data to be
15 output while data reading is discontinued by pickup's jump, continuous playback is realized.

Specifically, supposing the rate of reading the data from the DVD-RAM disk 331 is 11 Mbps, the maximum rate of outputting the data to the MPEG-2 decoding section 311 is 8
20 Mbps and the longest time it takes to move the pickup is 3

seconds, data of 24 megabits, which corresponds with the amount of data to be output to the MPEG-2 decoding section 311 while the pickup is moving, is needed as the extra output data. To secure this amount of data, the data needs to be
5 read for eight seconds on end. That is to say, the continuous reading needs to last for the amount of time that is obtained by dividing 24 megabits by the difference between the data read rate of 11 Mbps and the data output rate of 8 Mbps.

Accordingly, while the continuous reading is carried out
10 for eight seconds, data of 88 megabits, which should be output in eleven seconds, is read out. Thus, if a continuous data area with a size corresponding to at least eleven seconds is secured, then continuous data playback can be guaranteed.

15 It should be noted that several defective logical blocks may be included within the continuous data area. In that case, however, the continuous data area needs to have a size corresponding to an amount of time that is slightly longer than eleven seconds with the expected read time, which it will
20 take to read those defective logical blocks during the

playback operation, taken into account.

In performing the process of deleting a stored MP4 file, the writing control section 341 performs predetermined deletion processing by controlling the writing section 320 and reading section 321. In the MP4 file, the auxiliary information includes presentation timings (i.e., time stamps) of all frames. Accordingly, in partially deleting an intermediate portion of a moving picture stream, only the time stamps in the auxiliary information need to be deleted. It should be noted that in an MPEG-2 system stream, the moving picture stream should be analyzed to ensure continuity even at the partially deleted portion. This is because the time stamps are dispersed over the stream.

The MP4 file format is characterized by storing the video frames or audio frames of a video/audio stream as a single set without dividing each frame. At the same time, the MP4 file format defines access information, which enables random access to any arbitrary frames, for the first time ever for any international standard defined so far. The access information is defined on a frame-by-frame basis, and

may include the frame size, frame period, and address information for a frame. More specifically, the access information is stored for every unit (e.g., every display period of 1/30 second for a video frame and every 1,536 samples for an audio frame (in AC-3 audio, for example)). Accordingly, if the presentation timing of a video frame needs to be changed, only the access information thereof should be changed, and the video/audio stream does not always have to be changed. The access information of that type has a data size of about 1 megabyte per hour.

As to the data size of the access information, according to Non-Patent Document No. 1, the access information compliant with the DVD Video recording standard needs to have a data size of 70 kilobytes per hour. The data size of the access information as defined by the DVD Video recording standard is less than one-tenth of that of the access information included in the auxiliary information of an MP4 file. FIG. 8 schematically shows a correlation between the field names used as the access information compliant with the DVD Video recording standard and pictures represented by the field

names. FIG. 9 shows the data structure of the access information shown in FIG. 8, the field names defined for the data structure, and their contents and data sizes.

Also, the optical disk drive disclosed in Patent Document
5 No. 1 not only writes video frames on a GOP basis, not on a frame basis, but also writes each audio frame continuously for a period of time corresponding to one GOP. The optical disk drive defines the access information on a GOP basis, too, thereby cutting down the required data size of the access
10 information.

Furthermore, the MP4 file describes a moving picture stream in accordance with the MPEG-2 Video standard but is not compatible with a system stream as defined by the MPEG-2 System standard. Thus, the MP4 file cannot be edited by
15 utilizing the moving picture editing capability of any application used extensively today on PCs, for example. This is because the editing capability of a lot of applications is targeted on a moving picture stream compliant with the MPEG-2 System standard. Furthermore, the MP4 file standard defines
20 no decoder model to ensure playback compatibility for a moving

picture stream portion. Then, absolutely no piece of software or hardware compliant with the MPEG-2 System standard, which has circulated very widely today, can be used at all.

Meanwhile, a play list function for picking and combining
5 together preferred playback ranges of a moving picture file to make a single piece of work has been realized. This play list function is normally carried out as a virtual editing process without directly editing any moving picture file recorded. A play list is made up of MP4 files by newly generating a Movie
10 Atom. In making a play list up of MP4 files, if multiple playback ranges have the same stream attribute, then the same Sample Description Entry is used and the redundancy of Sample Description Entry can be reduced. However, when a seamless play list that guarantees seamless playback is described by
15 making use of this feature, it is difficult to describe the stream attribute information on a playback range basis.

An object of the present invention is to provide a data structure, of which the access information has a small data size and which can be used even in an application designed
20 for a conventional format, and also provide a data processor

that can perform processing based on such a data structure.

Another object of the present invention is to realize an editing process of combining video and audio seamlessly in a format compatible with a conventional stream that should have audio gaps. A more specific object thereof is to get such editing done on video and audio that is described as an MP4 stream. An additional object thereof is to combine audio naturally at every connection point.

Yet another object of the present invention is to realize an editing process that combines a plurality of contents together such that the user can specify his or her desired audio connection form (e.g., whether or not the audio should fade).

15 DISCLOSURE OF INVENTION

A data processor according to the present invention includes: a writing section for arranging a plurality of moving picture streams, each including video and audio to play back synchronously with each other, and writing the streams as at least one data file on a storage medium; and a

writing control section for locating a mute interval between two moving picture streams that are going to be played back continuously. The writing control section provides additional audio data representing audio to be reproduced in the mute interval located, and the writing section stores the provided additional audio data on the storage medium such that the additional audio data is associated with the data file.

The writing control section may further use audio data, which is stored in a predetermined terminal range of one of the two continuously played moving picture streams that is going to be played earlier than the other, and may provide the additional audio data including the same audio as that stored in the predetermined terminal range.

Alternatively, the writing control section may further use audio data, which is stored in a predetermined terminal range of one of the two continuously played moving picture streams that is going to be played later than the other, and may provide the additional audio data including the same audio as that stored in the predetermined terminal range.

The writing section may write the provided additional audio data just before where the mute interval is stored, thereby associating the additional audio data with the data file.

5 The writing section may write the arranged moving picture streams as a single data file on the storage medium.

Alternatively, the writing section may write the arranged moving picture streams as multiple data files on the storage medium.

10 The writing section may write the provided additional audio data just before where one of the two continuously played moving picture stream data files, which is going to be played later than the other, is stored, thereby associating the additional audio data with the data file.

15 The writing section may write information about the arrangement of the moving picture streams as at least one data file on the storage medium.

The mute interval may be shorter than the time length of a single audio decoding unit.

20 A video stream in each said moving picture stream may be

an MPEG-2 video stream, and the same MPEG-2 video stream
buffer conditions may have to be met by the two continuously
played moving picture streams.

The writing section may further write information for
5 controlling an audio level before and after the mute interval
on the storage medium.

The writing section may write the moving picture streams
in a physically continuous data area on the storage medium on
the basis of either a predetermined playback duration or a
10 predetermined data size, and may also write the additional
audio data just before the continuous data area.

A data processing method according to the present
invention includes the steps of: writing an arrangement of a
plurality of moving picture streams, each including video and
15 audio to play back synchronously with each other, as at least
one data file on a storage medium; and controlling writing by
locating a mute interval between two moving picture streams
that are going to be played back continuously. The step of
controlling writing includes providing additional audio data
20 representing audio to be reproduced in the mute interval

located, and the step of writing includes associating the provided additional audio data with the data file and storing the additional audio data on the storage medium.

The step of controlling writing may include further using
5 audio data, which is stored in a predetermined terminal range of one of the two continuously played moving picture streams that is going to be played earlier than the other, and providing the additional audio data including the same audio as that stored in the predetermined terminal range.

10 Alternatively, the step of controlling writing may include further using audio data, which is stored in a predetermined terminal range of one of the two continuously played moving picture streams that is going to be played later than the other, and providing the additional audio data
15 including the same audio as that stored in the predetermined terminal range.

The step of writing may include writing the provided additional audio data just before where the mute interval is stored, thereby associating the additional audio data with the
20 data file.

The step of writing may include writing the arranged moving picture streams as a single data file on the storage medium.

Alternatively, the step of writing may include writing
5 the arranged moving picture streams as multiple data files on the storage medium.

The step of writing may include writing the provided additional audio data just before where one of the two continuously played moving picture stream data files, which is
10 going to be played later than the other, is stored, thereby associating the additional audio data with the data file.

The step of writing may include writing information about the arrangement of the moving picture streams as at least one data file on the storage medium.

15 Another data processor according to the present invention includes: a reading section for reading at least one data file, including a plurality of moving picture streams with video and audio to be played back synchronously with each other, and additional audio data associated with
20 the at least one data file from a storage medium; a reading

control section for controlling reading by generating a control signal based on time information that is added to the moving picture streams to play back the video and audio synchronously with each other; and a decoding section for
5 decoding the moving picture streams in response to the control signal, thereby outputting signals representing the video and audio. When two moving picture streams are played back continuously by using the data processor, the reading control section outputs a control signal instructing that audio
10 represented by the additional audio data be output after one of two moving picture streams has been played back and before the other moving picture stream is played back.

Another data processing method according to the present invention includes the steps of: reading at least one data
15 file, including a plurality of moving picture streams with video and audio to be played back synchronously with each other, and additional audio data associated with the at least one data file from a storage medium; generating a control signal based on time information that is added to the moving
20 picture streams to play back the video and audio synchronously

with each other; and decoding the moving picture streams in response to the control signal, thereby outputting signals representing the video and audio. When two moving picture streams are played back continuously, the step of generating
5 the control signal includes outputting a control signal instructing that audio represented by the additional audio data be output after one of two moving picture streams has been played back and before the other moving picture stream is played back.

10 A computer program according to the present invention makes a computer function as a data processor that performs the following processing steps when read and executed by the computer. Specifically, by executing the computer program, the data processor performs the processing steps of:
15 acquiring a plurality of moving picture streams, each including video and audio to play back synchronously with each other, and writing the streams as at least one data file on a storage medium; and controlling writing by locating a mute interval between two moving picture streams that are
20 going to be played back continuously. The step of

controlling writing includes providing additional audio data representing audio to be reproduced in the mute interval located, and the step of writing includes associating the provided additional audio data with the data file and storing
5 the additional audio data on the storage medium.

The computer program may be stored on a storage medium.

Another data processor according to the present invention writes a plurality of encoded data, compliant with the MPEG-2 System standard, as a single data file such that
10 audio data of a predetermined length is associated with the data file.

Still another data processor according to the present invention reads a data file, including a plurality of encoded data compliant with the MPEG-2 System standard, and audio data
15 associated with the data file. In reading the encoded data, the data processor reads the audio data associated with the data file in a mute interval of the encoded data.

BRIEF DESCRIPTION OF DRAWINGS

20 FIG. 1 shows a configuration for a conventional data

processor 350.

FIG. 2 shows the data structure of an MP4 file 20.

FIG. 3 shows a specific example of the atom structure 23.

FIG. 4 shows the data structure of a moving picture
5 stream 22.

FIG. 5 shows a moving picture stream 22 in which tracks
are changed on the way.

FIG. 6 shows a correlation between a moving picture
stream 22 and sectors of a DVD-RAM disk 331.

10 FIG. 7 shows how the written data is managed by the file
system of the DVD-RAM.

FIG. 8 schematically shows a correlation between the
field names used as the access information compliant with the
DVD Video recording standard and pictures represented by the
15 field names.

FIG. 9 shows the data structure of the access information
shown in FIG. 8, the field names defined for the data
structure, and their contents and data sizes.

FIG. 10 illustrates a connection environment for a
20 portable videocorder 10-1, a camcorder 10-2 and a PC 10-3 for

carrying out the data processing of the present invention.

FIG. 11 shows an arrangement of functional blocks in a data processor 10.

FIG. 12 shows the data structure of an MP4 stream 12 according to the present invention.

FIG. 13 shows the management unit of audio data in an MPEG2-PS 14.

FIG. 14 shows a correlation between a program stream and elementary streams.

FIG. 15 shows the data structure of auxiliary information 13.

FIG. 16 shows the contents of respective atoms that make up an atom structure.

FIG. 17 shows a specific exemplary description format for "Data Reference Atom" 15.

FIG. 18 shows specific exemplary descriptions of respective atoms included in "Sample Table Atom" 16.

FIG. 19 shows a specific exemplary description format for "Sample Description Atom" 17.

FIG. 20 shows the contents of respective fields of

"sample_description_entry" 18.

FIG. 21 is a flowchart showing a procedure to generate the MP4 stream.

FIG. 22 is a table showing the differences between the
5 MPEG2-PS generated by the processing of the present invention and a conventional MPEG-2 Video (elementary stream).

FIG. 23 shows the data structure of the MP4 stream 12 in a situation where one VOBU is handled as one chunk.

FIG. 24 shows the data structure in the situation where
10 one VOBU is handled as one chunk.

FIG. 25 shows specific exemplary descriptions of respective atoms included in Sample Table Atom 19 in the situation where one VOBU is handled as one chunk.

FIG. 26 shows an exemplary MP4 stream 12 in which two PS
15 files are provided for a single auxiliary information file.

FIG. 27 shows an example in which there are a number of discontinuous MPEG2-PS's within one PS file.

FIG. 28 shows an MP4 stream 12 in which a PS file, storing an MPEG2-PS for the purpose of seamless connection, is
20 provided.

FIG. 29 shows the audio frame that is absent from the discontinuity point.

FIG. 30 shows the data structure of an MP4 stream 12 according to another example of the present invention.

5 FIG. 31 shows the data structure of an MP4 stream 12 according to still another example of the present invention.

FIG. 32 shows the data structure of an MTF file 32.

FIG. 33 shows a correlation among various types of file format standards.

10 FIG. 34 shows the data structure of a QuickTime stream.

FIG. 35 shows the contents of respective atoms in the auxiliary information 13 of the QuickTime stream.

FIG. 36 shows the contents of flags defined for a moving picture stream in a situation where the number of recording
15 pixels changes.

FIG. 37 shows the data structure of a moving picture file in which PS #1 and PS #3 are combined together so as to satisfy seamless connection conditions.

FIG. 38 shows conditions for seamlessly connecting video
20 and audio at the connection point between PS #1 and PS #3 and

playback timings thereof.

FIG. 39 shows a data structure in which an audio frame corresponding to an audio gap interval is allocated to a post recording area.

5 FIG. 40 shows audio overlap timings, in which portions (a) and (b) show two different overlap modes.

FIG. 41 shows playback timings in a situation where the playback ranges PS #1 and PS #3 are connected together so as to be played back seamlessly using a play list.

10 FIG. 42 shows the data structure of Sample Description Entry of the play list.

FIG. 43 shows the data structure of seamless information in Sample Description Entry of the play list.

FIG. 44 shows a seamless flag and STC continuity
15 information in a situation where seamless connection is done using a play list and a bridge file.

FIG. 45 shows the data structure of Edit List Atom of a PS track and an audio track in a play list.

FIG. 46 shows the data structure of Sample Description
20 Atom in the audio track in the play list.

BEST MODE FOR CARRYING OUT THE INVENTION

Hereinafter, preferred embodiments of the present invention will be described with reference to the accompanying
5 drawings.

FIG. 10 illustrates how to connect a portable videocorder 10-1, a camcorder 10-2 and a PC 10-3 for carrying out the data processing of the present invention.

The portable videocorder 10-1 receives a broadcast
10 program via its attached antenna and compresses the moving pictures of the broadcast program, thereby generating an MP4 stream. The camcorder 10-2 records not only video but also its accompanying audio, thereby generating another MP4 stream. In an MP4 stream, the video and audio data are encoded by a
15 predetermined compression coding method and are stored in accordance with the data structure described herein. The portable videocorder 10-1 and camcorder 10-2 either store the generated MP-4 streams on a storage medium 131 such as a DVD-RAM or output the streams through a digital interface such as
20 an IEEE 1394 or USB port. It should be noted that the

portable videocorder 10-1 and camcorder 10-2 needs to have even smaller sizes. Thus, the storage medium 131 does not have to be an optical disk with a diameter of 8 cm but may be an optical disk with a smaller diameter, for example.

5 The PC 10-3 receives the MP4 streams by way of either the storage medium or a transmission medium. If the respective appliances are connected together through a digital interface, then the PC 10-3 can receive the MP4 streams from the respective appliances by controlling the camcorder 10-2 and so
10 on as external storage devices.

 If the PC 10-3 has application software or hardware that can cope with the MP4 stream processing of the present invention, then the PC 10-3 can play back the MP4 streams just as defined by the MP4 file standard. On the other hand, if
15 the PC 10-3 cannot cope with the MP4 stream processing of the present invention, then the PC 10-3 can play back the moving picture streams in accordance with the MPEG-2 System standard. It should be noted that the PC 10-3 can also perform editing processing such as partial deletion on the MP4 streams. In
20 the following description, the portable videocorder 10-1,

camcorder 10-2 and PC 10-3 shown in FIG. 1 will be collectively referred to as a "data processor".

FIG. 11 shows an arrangement of functional blocks in a data processor 10. In the following description, the data processor 10 is supposed to have the capabilities of both reading and writing an MP4 stream. More specifically, the data processor 10 can not only generate an MP4 stream and write it on a storage medium 131 but also read an MP4 stream that is stored on the storage medium 131. The storage medium 131 may be a DVD-RAM disk, for example, and will be referred to herein as a "DVD-RAM disk 131".

First, the MP4 stream writing function of the data processor 10 will be described. The data processor 10 includes a video signal input section 100, an MPEG2-PS compressing section 101, an audio signal input section 102, an auxiliary information generating section 103, a writing section 120, an optical pickup 130 and a writing control section 141 as respective components regarding this function.

The video signal input section 100 is implemented as a video signal input terminal and receives a video signal

representing video data. The audio signal input section 102 is implemented as an audio signal input terminal and receives an audio signal representing audio data. For example, the video signal input section 100 and audio signal input section 102 of the portable videocorder 10-1 (see FIG. 10) may be connected to the video output section and audio output section of a tuner section (not shown) to receive a video signal and an audio signal, respectively. Also, the video signal input section 100 and audio signal input section 102 of the camcorder 10-2 (see FIG. 10) may respectively receive a video signal and an audio signal from the CCD output (not shown) and microphone output of a camera.

The MPEG2-PS compressing section (which will be simply referred to herein as a "compressing section") 101 receives the video and audio signals, thereby generating an MPEG-2 program stream (which will be referred to herein as an "MPEG2-PS") compliant with the MPEG-2 System standard. The MPEG2-PS generated may be decoded by itself in accordance with the MPEG-2 System standard. The MPEG2-PS will be described in further detail later.

The auxiliary information generating section 103 generates auxiliary information for the MP4 stream. The auxiliary information includes reference information and attribute information. The reference information is used to
5 identify the MPEG2-PS that has been generated by the compressing section 101 and may include the file name of the MPEG2-PS being written and its storage location on the DVD-RAM disk 131. On the other hand, the attribute information describes the attributes of a sample unit of the MPEG2-PS. As
10 used herein, the "sample" refers to the minimum management unit in a sample description atom (to be described later) as in the auxiliary information defined by the MP4 file standard. The attribute information includes data size, playback time and so on for each sample. One sample may be a data unit that
15 can be accessed at random, for example. In other words, the attribute information is needed to read the sample. Among other things, the sample description atom (to be described later) is sometimes called "access information".

Specific examples of the attribute information include
20 the address of the data storage location, a time stamp

representing playback timing, an encoding bit rate, and information about codec. The attribute information is provided for each of the video data and the audio data in every sample. Except for the field description to be
5 mentioned explicitly soon, the attribute information complies with the contents of auxiliary information for a conventional MP4 stream 20.

As will be described later, one sample according to the present invention is a single video object unit (VOBU) in the
10 MPEG2-PS. It should be noted that "VOBU" refers to the video object unit as defined by the DVD Video recording standard. The auxiliary information will be described in further detail later.

In accordance with the instruction given by the writing
15 control section 141, the writing section 120 controls the pickup 130, thereby writing data at a particular location (i.e., address) on the DVD-RAM disk 131. More specifically, the writing section 120 writes the MPEG2-PS, generated by the compressing section 101, and the auxiliary information,
20 generated by the auxiliary information generating section 103,

on the DVD-RAM disk 131 as respectively different files.

The data processor 10 further includes a continuous data area detecting section (which will be simply referred to herein as a "detecting section") 140 and a logical block management section (which will be simply referred to herein as a "management section") 143 that operate during the data write operation. In accordance with the instruction given by the writing control section 141, the continuous data area detecting section 140 checks the availability of sectors, which are managed by the logical block management section 143, thereby detecting a physically continuous unused area available. The writing control section 141 instructs the writing section 120 to write the data on that unused area. A specific data writing method may be similar to that already described with reference to FIG. 7, there is no particularly important difference, and the detailed description thereof will be omitted herein. It should be noted that the MPEG2-PS and the auxiliary information are written as separate files. Thus, their respective file names are written on the file identifiers shown in FIG. 7.

Hereinafter, the data structure of the MP4 stream will be described with reference to FIG. 12. FIG. 12 shows the data structure of an MP4 stream according to the present invention. The MP4 stream 12 includes an auxiliary information file (MOV001.MP4) including the auxiliary information 13 and a data file (MOV001.MPG) of the MPEG2-PS 14 (which will be referred to herein as a "PS file"). A single MP4 stream is made up of the data stored in these two files. In this description, the same name "MOV001" is given to the auxiliary information file and PS file to clearly indicate that these two files belong to the same MP4 stream but different extensions are given to them. More specifically, the same extension "MP4" as that of a conventional MP4 file is adopted as the extension of the auxiliary information file, while an extension "MPG" normally used in a conventional program stream is adopted as the extension of the PS file.

The auxiliary information 13 includes reference information **dref** to make reference to the MPEG2-PS 14 and further includes attribute information that describes the attributes of each video object unit (VOBU) of the MPEG2-PS 14.

Since the attribute information describes the attributes of each VOB, the data processor 10 can find a VOB at any arbitrary location in the MPEG2-PS 14 on a VOB basis and can perform playback and editing processing thereon.

5 The MPEG2-PS 14 is a moving picture stream, which is compliant with the MPEG-2 System standard and which is made up of video and audio packs that are interleaved together. Each video pack includes a pack header and encoded video data, while each audio pack includes a pack header and encoded audio
10 data. In the MPEG2-PS 14, the data is managed on a video object unit (VOBU) basis, where a VOB includes moving picture data, each unit of which has a length corresponding to a video playback duration of 0.4 to 1 second. The moving picture data includes a plurality of video packs and a plurality of audio
15 packs. By reference to the information described in the auxiliary information 13, the data processor 10 can locate and read any arbitrary VOB. It should be noted that each VOB includes at least one GOP.

The MP4 stream 12 of the present invention is partly
20 characterized in that the MPEG2-PS 14 can be decoded not only

by reference to the attribute information 13, which complies with the data structure of an MP4 stream as defined by the MPEG-4 system standard, but also in accordance with the MPEG-2 System standard. The auxiliary information file and the PS file are stored separately, and therefore, the data processor 5 can analyze and process them independently of each other. For example, an MP4 stream player, which can carry out the data processing of the present invention, can adjust the playback duration of the MP4 stream 12 according to the attribute information 13, sense the encoding method of the 10 MPEG2-PS 14 and decode it by its associated decoding method. On the other hand, a conventional apparatus that can decode an MPEG2-PS may decode it in accordance with the MPEG-2 System standard. Thus, even any currently popular version of 15 software or hardware, which is compliant with only the MPEG-2 System standard, can also play back a moving picture stream included in the MP4 stream.

Optionally, not only the sample description atom on the VOB basis but also another sample description atom, which 20 uses a number of frames of the audio data of the MPEG2-PS 14,

corresponding to a predetermined amount of time, as a management unit, may be provided as shown in FIG. 13. The predetermined amount of time may be 0.1 second, for example. In FIG. 13, "V" denotes the video pack shown in FIG. 12 and "A" denotes the audio pack. An audio frame corresponding to 0.1 second is made up of at least one pack. As for AC-3, for example, one audio frame includes audio data corresponding to 1,536 samples supposing the sampling frequency is 48 kHz. In this case, the sample description atom may be provided either within a user data atom in a track atom or on an independent track. In another example, the auxiliary information 13 may use an audio frame, synchronized with a VOB and corresponding to a duration of 0.4 second to 1 second, as a unit and may store various attributes such as the overall data size of the units, the data address of the top pack and a time stamp representing the output timing.

Next, the data structure of each video object unit (VOBU) of the MPEG2-PS 14 will be described. FIG. 14 shows a correlation between a program stream and elementary streams. In the MPEG2-PS 14, a single VOB includes a plurality of

video packs V_PCK and a plurality of audio packs A_PCK. More exactly, a VOB runs from a sequence header (i.e., SEQ header shown in FIG. 14) to a pack just before the next sequence header. That is to say, a sequence header is put at the top of each VOB. On the other hand, the elementary stream (Video) includes a number N of GOPs, which include various types of headers (such as the sequence (SEQ) header and GOP header) and video data (including I-, P- and B-pictures). The elementary stream (Audio) includes a plurality of audio frames.

Each of the video and audio packs included in the VOB of the MPEG2-PS 14 is composed of the data included in its associated elementary stream (Video) or (Audio) so as to have a data size of 2 kilobytes. As described above, each pack is provided with a pack header.

It should be noted that if another elementary stream (not shown) is provided for auxiliary video data such as subtitle data, then each VOB of the MPEG2-PS 14 further includes packs of that auxiliary video data.

Next, the data structure of the auxiliary information 13 in the MP4 stream 12 will be described with reference to FIGS.

15 and 16. FIG. 15 shows the data structure of the auxiliary
information 13. This data structure is called an "atom
structure" and has a layered architecture. For example,
"Movie Atom" includes "Movie Header Atom", "Object Descriptor
5 Atom" and "Track Atom", which is further subdivided into
"Track Header Atom", "Edit List Atom", "Media Atom" and "User
Data Atom". A similar statement applies to the other Atoms
shown in FIG. 15.

According to the present invention, the attributes of a
10 sample unit are described by using "Data Reference Atom (drf)"
15 and "Sample Table Atom (stbl)" 16, in particular. As
described above, one sample corresponds to one video object
unit (VOBU) of the MPEG2-PS. "Sample Table Atom" 16 includes
the six low-order atoms shown in FIG. 15.

15 FIG. 16 shows the contents of respective atoms that make
up the atom structure. "Data Reference Atom" stores the
information identifying the file of a moving picture stream
(i.e., the MPEG2-PS) 14 in the form of a URL. On the other
hand, "Sample Table Atom" describes the attributes of
20 respective VOBUs with its low-order atoms. For example,

"Decoding Time to Sample Atom" stores the playback durations of the respective VOBUs. "Sample Size Atom" stores the data sizes of the respective VOBUs. Also, "Sample Description Atom" shows that the PS file data making up the MP4 stream 12 is the MPEG2-PS 14 and also provides detailed specifications of the MPEG2-PS 14. In the following description, the information described by "Data Reference Atom" will be referred to herein as "reference information" and the information described by "Sample Table Atom" will be referred to herein as "attribute information".

FIG. 17 shows a specific exemplary description format for "Data Reference Atom" 15. The information identifying the file is described in a portion ("DataEntryUrlAtom" in this example) of the field describing "Data Reference Atom" 15. In this case, the file name and file storage location of the MPEG2-PS 14 are described as a URL. By reference to "Data Reference Atom" 15, the MPEG2-PS 14, which makes up the MP4 stream 12 along with its auxiliary information 13, can be identified. It should be noted that even before the MPEG2-PS 14 is written on the DVD-RAM disk 131, the auxiliary

information generating section 103 shown in FIG. 11 can also detect the file name and file storage location of the MPEG2-PS 14. This is because the file name can be determined in advance and because the file storage location can be logically 5 identified by the notation of the layered structure of the file system.

FIG. 18 shows specific exemplary descriptions of respective atoms included in "Sample Table Atom" 16. Each atom defines the field name, repeatability and data size. For 10 example, "Sample Size atom" includes three fields "sample-size", "sample count" and "entry-size". Among these fields, the default data size of the VOB is stored in the "sample-size" field, and an individual data size, which is different from the default value of the VOB, is stored in the "entry-size" 15 field. In the "setting" shown in FIG. 18, each parameter (such as "VOB_ENT") may have the same value as the access data of the same name according to the DVD Video standard.

In "Sample Description Atom" 17 shown in FIG. 18, the 20 attribute information of the sample unit is described.

Hereinafter, the contents of the information described in "Sample Description Atom" 17 will be described.

FIG. 19 shows a specific exemplary description format for "Sample Description Atom" 17. "Sample Description Atom" 17 describes its data size and the attribute information of a sample unit when each VOB is a single sample. The attribute information is described in "sample_description_entry" 18 of "Sample Description Atom" 0.

FIG. 20 shows the contents of respective fields of "sample_description_entry" 18. The entry 18 includes "data format" specifying the encoding method of its associated MPEG2-PS 14. In FIG. 20, "p2sm" shows that the MPEG2-PS 14 is an MPEG-2 program stream including MPEG-2 Video.

The entry 18 includes that sample's "Presentation Start Time" and "Presentation End Time", which store the timing information of the first video frame and the timing information of the last video frame, respectively. The entry 18 further includes the attribute information ("Video ES Attribute") of the video stream within the sample and the attribute information ("Audio ES Attribute") of the audio

stream within the same sample. As shown in FIG. 19, the video data attribute information may define the CODEC type of the video (e.g., MPEG-2 Video) and the width and height ("width" and "height") of the video data, for example. In the same way, 5 the audio data attribute information may define the CODEC type of the audio (e.g., AC-3), the number of channels of the audio data ("channel count"), the size of the audio sample ("samplesize") and the sampling rate thereof ("samplerate").

The entry 18 further includes a discontinuity point start 10 flag and seamless information. These pieces of information are described if there are a number of PS streams in a single MP4 stream 12 as will be described later. For example, a discontinuity point start flag of "0" indicates that the previous moving picture stream and the current moving picture 15 stream are a completely continuous program stream. On the other hand, a discontinuity point start flag of "1" shows that those moving picture streams are discontinuous program streams. And if those streams are discontinuous, the seamless information may be described in order to play back a moving 20 picture or audio without a break even at a discontinuity point

of the moving picture or audio. The seamless information includes audio discontinuity information and SCR discontinuity information during the playback. The audio discontinuity information includes the presence or absence of a mute interval (i.e., the audio gap shown in FIG. 31), the start timing and the time length thereof. The SCR discontinuity information includes the SCR values of the two packs that are just before, and just after, the discontinuity point.

By providing the discontinuity point start flag, the switching point of Sample Description Entries and the continuity switching point of moving picture streams can be defined independently of each other. As shown in FIG. 36, if the number of recording pixels changes on the way, then Sample Descriptions are changed. In this case, however, if the moving picture streams themselves are continuous, then the discontinuity point start flag may set to zero. If the discontinuity point start flag is zero, a PC, which is directly editing an information stream, can understand that seamless playback is realized even without resetting a connection point between two moving picture streams. FIG. 36

shows a situation where the number of horizontal pixels has changed. However, the same technique is also applicable to a situation where any other type of attribute information has changed. For example, a situation where a 4:3 aspect ratio
5 has changed into 16:9 as to the aspect information or a situation where the audio bit rate has changed may also be coped with.

The data structures of the auxiliary information 13 and MPEG2-PS 14 of the MP4 stream 12 shown in FIG. 12 have been
10 described. According to the data structure described above, any portion of the MPEG2-PS 14 may be deleted just by changing the attribute information (e.g., the time stamp) in the auxiliary information 13, and there is no need to change the time stamp provided for the MPEG2-PS 14. Thus, the editing
15 can be done by taking advantage of a conventional MP4 stream. In addition, according to the data structure described above, if a moving picture is being edited on a PC with application or hardware compatible with a stream compliant with the MPEG-2 System standard, just a PS file may be imported into the PC.
20 This is because the MPEG2-PS 14 of the PS file is a moving

picture stream compliant with the MPEG-2 System standard. Such application or hardware has circulated widely, and therefore, any piece of existent software or hardware can be used effectively. In addition, the auxiliary information can
5 be stored in a data structure compliant with the ISO standard.

Hereinafter, it will be described with reference to FIGS. 11 and 21 how the data processor 10 generates an MP4 stream and writes it on a DVD-RAM disk 131. FIG. 21 is a flowchart showing a procedure to generate the MP4 stream. First, in
10 Step 210, the data processor 10 receives video data through the video signal input section 100 and audio data through the audio signal input section 102, respectively. Next, in Step 211, the compressing section 101 encodes the received video and audio data in accordance with the MPEG-2 System standard.
15 Subsequently, in Step 212, the compressing section 101 makes up an MPEG2-PS of the video and audio encoded streams (see FIG. 14).

Then, in Step 213, the writing section 120 determines the file name and storage location of the MPEG2-PS to be written
20 on the DVD-RAM disk 131. Next, in Step 214, the auxiliary

information generating section 103 acquires the file name and storage location of the PS file and specifies the contents to be described as the reference information (i.e., Data Reference Atom shown in FIG. 17). As shown in FIG. 17, a description method that makes it possible to specify the file name and the storage location at the same time is adopted herein.

Subsequently, in Step 215, the auxiliary information generating section 103 acquires data representing the playback duration, data size and so on for each of the VOBUs defined in the MPEG2-PS 14 and specifies the contents to be described as the attribute information (i.e., Sample Table Atom shown in FIGS. 18 through 20). By providing the attribute information for each VOB, any arbitrary VOB can be read and decoded. This means that one VOB is handled as one sample.

Thereafter, in Step 216, the auxiliary information generating section 103 generates the auxiliary information based on the reference information (i.e., Data Reference Atom) and the attribute information (i.e., Sample Table Atom).

Next, in Step 217, the writing section 120 outputs the

auxiliary information 13 and MPEG2-PS 14 as the MP4 stream 12
and writes them on the DVD-RAM disk 131 as an auxiliary
information file and a PS file, respectively. By performing
this procedure, the MP4 stream is generated and written on the
5 DVD-RAM disk 131.

Hereinafter, the MP4 stream reading function of the data
processor 10 will be described with reference to FIGS. 11 and
12. On the DVD-RAM disk 131, the MP4 stream 12, including the
auxiliary information 13 and MPEG2-PS 14 having the data
10 structures described above, is supposed to be stored. Upon a
user's request, the data processor 10 reads and decodes the
MPEG2-PS 14 that is stored on the DVD-RAM disk 131. The data
processor 10 includes a video signal output section 110, an
MPEG2-PS decoding section 111, an audio signal output section
15 112, a reading section 121, the pickup 130 and a reading
control section 142 as respective components realizing the
reading function.

First, in accordance with an instruction given by the
reading control section 142, the reading section 121 controls
20 the pickup 130 so as to read the MP4 file from the DVD-RAM

disk 131 and acquire the auxiliary information 13. The reading section 121 outputs the acquired auxiliary information 13 to the reading control section 142. Also, in response to a control signal supplied from the reading control section 142 to be described later, the reading section 121 reads the PS file from the DVD-RAM disk 131. The control signal is a signal designating the PS file to read ("MOV001.MPG").

The reading control section 142 receives the auxiliary information 13 from the reading section 121 and analyzes its data structure, thereby acquiring the reference information 15 (see FIG. 17) contained in the auxiliary information 13. Then, the reading control section 142 outputs a control signal instructing that the PS file ("MOV001.MPG") designated by the reference information 15 be read from the specified location (i.e., "./", or root directory).

The MPEG2-PS decoding section 111 receives the MPEG2-PS 14 and the auxiliary information 13 and decodes the MPEG2-PS 14 into video data and audio data in accordance with the attribute information contained in the auxiliary information 13. More specifically, the MPEG2-PS decoding section 111

reads the data format ("data_format"), the video stream attribute information ("video ES attribute") and the audio stream attribute information ("audio ES attribute") of Sample Description Atom 17 (see FIG. 19), and decodes the video and audio data in accordance with the encoding method, the presentation size of the video data and the sampling frequency as defined by those pieces of information.

The video signal output section 110 is implemented as a video signal output terminal to output the decoded video data as a video signal, while the audio signal output section 112 is implemented as an audio signal output terminal to output the decoded audio data as an audio signal.

The MP4 stream reading process by the data processor 10 begins by reading a file with an extension "MP4" (i.e., "MOV001.MP4") as in the conventional process of reading an MP4 stream file. More specifically, this process may be carried out in the following manner. First, the reading section 121 reads the auxiliary information file ("MOV001.MP4"). Next, the reading control section 142 analyzes the auxiliary information 13, thereby extracting the reference information

(i.e., Data Reference Atom). Then, in accordance with the reference information extracted, the reading control section 142 outputs a control signal instructing that the PS file, making up the same MP4 stream, be read. In this preferred embodiment, the control signal output by the reading control section 142 instructs that the PS file ("MOV001.MPG") be read.

Next, in response to the control signal, the reading section 121 reads the designated PS file. Thereafter, the MPEG2-PS decoding section 111 receives the MPEG2-PS 14 and auxiliary information 13 contained in the data file read and analyzes the auxiliary information 13, thereby extracting the attribute information. Then, by reference to Sample Description Atom 17 (see FIG. 19) included in the attribute information, the MPEG2-PS decoding section 111 identifies the data format of the MPEG2-PS 14 ("data_format"), the attribute information of the video stream included in the MPEG2-PS 14 ("video ES attribute") and the attribute information of the audio stream ("audio ES attribute"), thereby decoding the video data and audio data. By performing these processing steps, the MPEG2-PS 14 is read in accordance with the

auxiliary information 13.

It should be noted that any conventional player or playback software that can read a stream compliant with the MPEG-2 System standard can read the MPEG2-PS 14 just by
5 reading the PS file. In that case, however, the player does not have to be able to read the MP4 stream 12. Since the MP4 stream 12 is made up of the auxiliary information 13 and MPEG2-PS 14 as two separate files, the PS file in which the MPEG2-PS 14 is stored can be easily identified by the
10 extension, for example, and read.

FIG. 22 is a table showing the differences between the MPEG2-PS generated by the processing of the present invention and a conventional MPEG-2 Video (elementary stream). In FIG. 22, the column "the present invention (1)" summarizes the
15 above example in which one VOB unit is handled as one sample. In the conventional example, one video frame is handled as one sample and attribute information (access information) such as Sample Table Atom is provided for each sample. In contrast, according to the present invention, a VOB unit including a
20 plurality of video frames is used as a sample unit and the

access information is provided for each sample, thus cutting down the amount of attribute information significantly. That is why one VOBUs is preferably treated as one sample as in the present invention.

5 In FIG. 22, the column "the present invention (2)" shows a modified example of the data structure of "the present invention (1)". The difference between "the present invention (2)" and "the present invention (1)" lies in that in this modified example (i.e., the present invention (2)), one VOBUs
10 corresponds to one chunk and that the access information is defined on a chunk-by-chunk basis. As used herein, one "chunk" is a unit consisting of a plurality of samples. In this example, a video frame including the pack header of the MPEG2-PS 14 corresponds to one sample. FIG. 23 shows the data
15 structure of the MP4 stream 12 in a situation where one VOBUs is handled as one chunk. The difference is that each sample shown in FIG. 12 is replaced by one chunk. In the conventional example, one video frame is handled as one sample and one GOP is treated as one chunk.

20 FIG. 24 shows the data structure in the situation where

one VOB is handled as one chunk. Comparing this data structure with that shown in FIG. 15 in which one VOB is treated as one sample, it can be seen that the contents defined by Sample Table Atom 19 included in the attribute information of the auxiliary information 13 are different. FIG. 25 shows specific exemplary descriptions of respective atoms included in Sample Table Atom 19 in the situation where one VOB is handled as one chunk.

Hereinafter, a modified example of the PS file to make up the MP4 stream 12 will be described. FIG. 26 shows an exemplary MP4 stream 12 in which two PS files ("MOV001.MPG" and "MOV002.MPG") are provided for a single auxiliary information file ("MOV001.MPG"). In these two PS files, the data of the MPEG2-PS 14, representing mutually different moving picture scenes, are stored separately. Within each PS file, the moving picture stream is continuous, and each of a system clock reference (SCR), a presentation time stamp (PTS) and a decoding time stamp (DTS), which are all compliant with the MPEG-2 System standard, is continuous, too. However, the SCR's, PTS's and DTS's are not continuous with each other

between the two PS files (i.e., between the end of MPEG2-PS #1 included in one PS file and the beginning of MPEG2-PS #2 included in the other PS file). These two PS files are treated as separate tracks (diagrams).

5 In the auxiliary information file, reference information (dref, see FIG. 17) for identifying the file names and storage locations of the respective PS files is described. The reference information may be described in the order of items to be referred to, for example. In FIG. 26, the PS file
10 "MOV001.MPG" identified by Reference #1 is read first, and then the PS file "MOV002.MPG" identified by Reference #2 is read. Even if there are a number of PS files in this manner, those PS files can be read substantially continuously by providing reference information for the respective PS files
15 within the auxiliary information file.

FIG. 27 shows an example in which there are a number of discontinuous MPEG2-PS's within one PS file. In the PS file, MPEG2-PS's #1 and #2, representing different moving picture scenes, are arranged back to back. The "discontinuous MPEG2-
20 PS's" mean that the SCR's, PTS's and DTS's are not continuous

with each other between the two MPEG2-PS's (i.e., between the end of MPEG2-PS #1 and the beginning of MPEG2-PS #2). In other words, it means that the read timings are not continuous with each other. The discontinuity point is located in the boundary between the two MPEG2-PS's. It should be noted that within each MPEG2-PS, the moving picture stream is continuous and each of SCR, PTS and DTS, which are all compliant with the MPEG-2 System standard, is continuous, too.

In the auxiliary information file, reference information (dref, see FIG. 17) for identifying the file name and storage location of the PS file is described. A single piece of reference information designating the PS file is stored in the auxiliary information file. However, if that PS file were read sequentially, then the read operation would stop at the discontinuity point between MPEG2-PS #1 and MPEG2-PS #2 because the SCR's, PTS's and DTS's are discontinuous with each other there. Thus, information about this discontinuity point (e.g., location information (or address) of the discontinuity point) is described in the auxiliary information file. More specifically, the location information of the discontinuity

point is stored as the "discontinuity point start flag" shown in FIG. 19. For example, during the read operation, the reading control section 142 detects the location information of the discontinuity point and reads the video data of MPEG2-PS #2, which is located after the discontinuity point, in advance, thereby controlling the read operation such that at least the video data can be played back without a break.

A procedure of reading two PS files, storing mutually discontinuous MPEG2-PS's, by providing two pieces of reference information for the files has been described with reference to FIG. 26. Optionally, as shown in FIG. 28, another PS file, storing an MPEG2-PS for the purpose of seamless connection, may be newly inserted between the two PS files such that the two original PS files can be read seamlessly. FIG. 28 shows an MP4 stream 12 in which a PS file ("MOV002.MPG"), storing an MPEG2-PS for the purpose of seamless connection, is provided. The PS file ("MOV002.MPG") includes an audio frame that is absent from the discontinuity point between MPEG2-PS #1 and MPEG2-PS #3. This point will be described in further detail with reference to FIG. 29.

FIG. 29 shows the audio frame that is absent from the discontinuity point. In FIG. 29, the PS file storing MPEG2-PS #1 is identified by "PS #1" and the PS file storing MPEG2-PS #3 is identified by "PS #3".

5 Suppose the data of PS #1 is processed first, and then that of PS #3 is processed. The DTS video frame on the second row and the PTS video frame on the third row represent time stamps of a video frame. As can be seen from these time stamps, PS files #1 and #3 can be played back without
10 discontinuing the video. As to an audio frame, however, there is a mute interval, in which no data is present for a certain period of time, after PS #1 has been played and before PS #3 starts being played. With such an interval left, seamless playback could not be achieved.

15 Thus, PS #2 is newly provided and a PS file, including an audio frame for the purpose of seamless connection, is provided such that the auxiliary information file can make reference to that file. This audio frame includes audio data to fill the mute interval. For example, the audio data that
20 was written synchronously with the end of the moving picture

of PS #1 is copied. As can be seen from the audio frame row shown in FIG. 29, the audio frame for the purpose of seamless connection is inserted next to PS #1. The audio frame of PS #2 lasts until less than one frame before PS #3 begins. Accordingly, another piece of reference information (dref shown in FIG. 28) to make reference to this new PS #2 is provided for the auxiliary information 13 and is defined such that PS #2 is referred to after PS #1.

In FIG. 29, no data interval for less than one audio frame (i.e., a mute interval) is shown as "audio gap". Alternatively, the mute interval may be eliminated by adding extra data for one more audio frame to PS #2. In that case, PS #2 and PS #3 will include a portion with the same audio data sample, i.e., a portion in which the audio frames overlap with each other. Even so, no serious problem should arise. This is because as to the overlapping portion, the same sound will be output no matter which data is read.

In the moving picture streams PS #1 and PS #3, the video stream thereof preferably satisfies the VBV buffer condition of the MPEG-2 Video standard continuously before and after

its connection point. This is because if the buffer condition is satisfied, no underflow will occur in the video buffer in the MPEG2-PS decoding section, and therefore, the reading control section 142 and MPEG2-PS decoding section 111 can easily play back the video seamlessly.

By performing these processing steps, even a number of discontinuous PS files can be read and decoded continuously with no time gap left.

In the example shown in FIG. 29, all PS files are supposed to be referred to by the reference information **dref**. However, just the PS #2 file may be referred to by any other atom (e.g., a uniquely defined dedicated atom) or the second PS track. In other words, only the PS files compliant with the DVD Video recording standard may be referred to by the **dref** atom. Alternatively, the audio frame in the PS #2 file may be stored as an independent file for the elementary stream, may be referred to by an independent audio track atom provided within the auxiliary information file, and may be described in the auxiliary information file so as to be played back in parallel with the end of PS #1. The timing of playing back PS

#1 and the audio elementary stream simultaneously may be specified by Edit List Atom (see FIG. 15, for example) in the auxiliary information.

In the preferred embodiments described above, the moving picture stream is supposed to be an MPEG-2 program stream. Alternatively, the moving picture stream may also be an MPEG-2 transport stream (which will be referred to herein as an "MPEG2-TS") as defined by the MPEG-2 System standard.

FIG. 30 shows the data structure of an MP4 stream 12 according to another example of the present invention. The MP4 stream 12 includes an auxiliary information file ("MOV001.MP4"), storing auxiliary information 13, and the data file ("MOV001.M2T") of an MPEG2-TS 14 (which will be referred to herein as a "TS file").

As in the MP4 stream 12 shown in FIG. 12, the TS file is also referred to by the reference information *dref* in the auxiliary information 13 in this MP4 stream 12.

A time stamp is added to the MPEG2-TS 14. More specifically, in this MPEG2-TS 14, a time stamp of 4 bytes to be referred to at the time of transmission is additionally

provided before a transport packet (which will be referred to herein as a "TS packet") of 188 bytes. Accordingly, a TS packet containing video (V_TSP) and a TS packet containing audio (A_TSP) are each made up of 192 bytes. It should be
5 noted that the time stamp may be provided behind the TS packet.

In the MP4 stream 12 shown in FIG. 30, the attribute information may be described in the auxiliary information 13 with a TS packet, containing video data corresponding to a video playback duration of about 0.4 second to about 1 second,
10 treated as one sample as in the VOB shown in FIG. 12. In addition, as in FIG. 13, the data size, data address and playback timing of one frame of audio data may also be described in the auxiliary information.

Alternatively, one frame may be handled as one sample,
15 and a plurality of frames may be treated as one chunk. FIG. 31 shows the data structure of an MP4 stream 12 according to still another example of the present invention. In this case, if a plurality of TS packets, each containing video data corresponding to a video playback duration of about 0.4 second
20 to about 1 second, is handled as one chunk and if access

information is defined on a chunk-by-chunk basis as in FIG. 23, then quite the same effects as those achieved by the MP4 stream 12 shown in FIG. 12 are also accomplished.

Even if the data structure shown in FIG. 30 or 31 is
5 adopted, the arrangement of respective files and the processing to be carried out based on the data structure are similar to those already described with reference to FIGS. 12, 13 and 23. Thus, the description thereof will be omitted herein because it is easily understandable just by applying
10 the statements about the video and audio packs shown in FIGS. 12, 13 and 23 to the video TS packet with the time stamp (V_TSP) and the audio TS packet with the time stamp (A_TSP) shown in FIG. 30.

Next, the file structure of another data format, to which
15 the data processing described above is also applicable, will be described with reference to FIG. 32. FIG. 32 shows the data structure of an MTF file 32. The MTF 32 is a file for storing a written or edited moving picture. The MTF file 32 includes a plurality of continuous MPEG2-PS's 14, while each
20 MPEG2-PS 14 includes a plurality of samples ("P2Sample").

Every sample ("P2Sample") is one continuous stream. For example, as already described with reference to FIG. 12, the attribute information may be defined on a sample basis. In the foregoing description, this sample ("P2Sample")
5 corresponds to a VOB. Every sample includes a plurality of video and audio packs, each of which contains a constant quantity of data of 2,048 bytes. Also, if two MTFs are combined together, then the resultant MTF will consist of at least two P2Stream's.

10 In the MTF 32, if two adjacent MPEG2-PS's 14 are one continuous program stream, then a single piece of reference information may be provided for the continuous range, thereby making up one MP4 stream. On the other hand, if two adjacent MPEG2-PS's 14 are a discontinuous program stream, then the
15 data address of the discontinuity point may be included in the attribute information as shown in FIG. 27, thereby making up another MP4 stream 12. Thus, the data processing described above is applicable to the MTF 32, too.

It has been described how to handle an MPEG-2 system
20 stream by extending the MP4 file format that was standardized

in 2001. Alternatively, according to the present invention, the MPEG-2 system stream may also be handled even by extending the QuickTime file format or the ISO Base Media file format, too. This is because most of the specifications of the MP4
5 file format and the ISO Base Media file format are defined based on, and have the same contents as, the QuickTime file format. FIG. 33 shows a correlation among various types of file format standards. For a type of atom (moov, mdat) in which "the present invention", "MP4 (2001)" and "QuickTime"
10 overlap with each other, the data structure of the present invention described above can be adopted. As already described, the atom type "moov" is shown in FIG. 15 and other drawings as "Movie Atom" of the highest-order layer of the auxiliary information.

15 FIG. 34 shows the data structure of a QuickTime stream. The QuickTime stream also consists of a file ("MOV001.MOV") describing the auxiliary information 13 and a PS file ("MOV001.MPG") including the MPEG2-PS 14. Compared with the MP4 stream 12 shown in FIG. 15, "Movie Atom" defined by the
20 auxiliary information 13 of the QuickTime stream is partially

changed. Specifically, "Null Media Header Atom" is replaced with "Base Media Header Atom" 36 newly provided, and "Object Descriptor Atom" shown on the third row of FIG. 15 is deleted from the auxiliary information 13 shown in FIG. 34. FIG. 35 shows the contents of respective atoms in the auxiliary information 13 of the QuickTime stream. If the data of a sample (VOBU) is neither a video frame nor an audio frame, the "Base Media Header Atom" 36 added indicates that. The other atom structure shown in FIG. 35 and its contents are the same as those of the MP4 stream 12 described above and the description thereof will be omitted herein.

Hereinafter, audio processing for realizing a seamless playback will be described. First, a conventional seamless playback will be described with reference to FIGS. 37 and 38.

FIG. 37 shows the data structure of a moving picture file in which PS #1 and PS #3 are combined together so as to satisfy seamless connection conditions. In the moving picture file MOVE0001.MPG, two continuous moving picture streams PS #1 and PS #3 are connected together. The moving picture file has a playback duration of a predetermined length (of 10 seconds

to 20 seconds, for example). A post recording data area is provided physically just before the moving picture streams of the predetermined length. In this data area, a post recording empty area, which is an unused area, is provided as a separate
5 file named MOVE0001.EMP.

It should be noted that if the moving picture file has a longer playback duration, then there will be multiple sets of post recording areas and moving picture stream areas of a predetermined length. If these sets are written continuously
10 on a DVD-RAM disk, then the moving picture file is stored so as to be interleaved with those post recording areas. This format is adopted to make the data stored in any of those post recording areas easily accessible in a short time even while the moving picture file is being accessed.

15 Also, the video streams in the moving picture file are supposed to satisfy the VBV buffer conditions as defined by the MPEG-2 Video standard continuously before and after the connection point between PS #1 and PS #3 (as well as the connection conditions for realizing a seamless playback at a
20 connection point between two streams as defined by the DVD-VR

standard).

FIG. 38 shows conditions for seamlessly connecting video and audio at the connection point between PS #1 and PS #3 shown in FIG. 37 and playback timings thereof. An extended
5 portion of the audio frame to be reproduced synchronously with the last video frame of PS #1 is stored at the top of PS #3. There is an audio gap between PS #1 and PS #3. This audio gap is the same as that already described with reference to FIG. 29. In FIG. 29, if the video of PS #1 and the video of PS #3
10 are played back continuously without a break, then the audio gap will be produced because the audio frame playback period of PS #1 becomes different from that of PS #3. This phenomenon is caused because the playback period of a video frame does not match that of an audio frame. A conventional
15 player stops reproducing audio in this audio gap interval. As a result, the audio reproduction discontinues, although just a moment, at the stream connection point.

To avoid such audio discontinuity, a fade-out, fade-in measure may be taken before and after the audio gap. That is
20 to say, by applying the fade-out and fade-in in 10 ms each

before and after the audio gap of a seamless playback, noise to be caused by the sudden audio stoppage can be eliminated and the audio can be heard more natural. However, if the fade-out and fade-in were applied every time the audio gap is produced, then stabilized audio level could not be provided depending on the type of the audio material in question, and good audiovisual state could not be maintained anymore. That is why it is sometimes necessary to eliminate the mute range caused by the audio gap during the playback.

For that purpose, the following measure is taken in this preferred embodiment. FIG. 39 shows the physical data arrangement of a moving picture file MOVE0001.MPG and an audio file OVRP0001.AC3 in a situation where the audio frame OVRP0001.AC3, which can fill the audio gap interval, is written in a portion of a post recording data area. These moving picture and audio files are generated by the writing section 120 in accordance with an instruction (or control signal) given by the writing control section 141.

To make such a data arrangement, the writing control section 141 realizes a seamlessly reproducible data structure,

which allows an audio gap, for the data that is located around the connection point between the moving picture streams PS #1 and PS #3 that should be connected seamlessly together. At this point in time, it is known whether or not there is any
5 no-data interval (i.e., a mute interval) corresponding to one audio frame or less (i.e., whether or not there is an audio gap), and an audio frame including the audio data to be lost in that audio gap interval and the length of the audio gap interval are also known. The audio gap is produced in almost
10 all cases. Next, the writing control section 141 transmits the data of the audio that should be reproduced in that audio gap interval to the writing section 120 and makes the writing section 120 store it as an audio file and associate it with the moving picture file. As used herein, "to associate" means
15 providing a post recording data area just before the moving picture file is stored and storing additional audio data in that data area. It also means associating that moving picture file and a file storing the audio data with a moving picture track and an audio track, respectively, in the auxiliary
20 information (Movie Atom). This audio data may be audio frame

data compliant with the AC3 format, for example.

As a result, the moving picture data file shown in FIG. 39 (i.e., MOVE0001.MPG and OVRP0001.AC3) is written on the DVD-RAM disc 131. It should be noted that the unused portion of the post recording data area should be reserved as a different file MOVE0001.EMP.

FIG. 40 shows audio overlap playback timings in two different modes. Portion (a) of FIG. 40 shows a first overlap mode, while portion (b) of FIG. 40 shows a second overlap mode. Specifically, portion (a) of FIG. 40 shows a mode in which the playback range of the audio frame OVRP0001.AC3 overlaps with that of the top frame of PS #3 that is located right after the audio gap. The overlapping audio frame is registered as an audio track in the auxiliary information in the moving picture file. Also, the playback timing of this overlapping audio frame is recorded as an Edit List Atom for an audio track in the auxiliary information of the moving picture file. However, it depends on the playback processing done by the data processor 10 how to reproduce the two overlapping audio ranges. For example, in accordance with the

instruction given by the reading control section 142, the reading section 121 reads OVRP0001.AC3 first and then PS #2 and #3 from the DVD-RAM in this order. In the meantime, the MPEG2-PS decoding section 111 starts playing back PS #2.

5 After having played back PS #2, the MPEG2-PS decoding section 111 starts playing back the top of PS #3 and reads its audio frame at the same time. Thereafter, when the reading section 121 reads the audio frame of PS #3, the MPEG2-PS decoding section 111 delays its playback timing by the amount of

10 overlap and then starts playing it. However, if the playback timing were delayed at every connection point, then the video-audio gap might broaden to a sensible degree. That is why it is necessary to read and output the audio frame of PS #3 at its original playback timing without using OVRP0001.AC3 all

15 through the playback range.

On the other hand, portion (b) of FIG. 40 shows a mode in which the playback range of the audio frame OVRP0001.AC3 overlaps with that of the last frame of PS #3 that is located just before the audio gap. In this mode, in accordance with

20 the instruction given by the reading control section 142, the

reading section 121 reads the overlapping audio frame first and then the audio frames of PS #2 and #3 in this order. When the reading section 121 starts reading PS #2, the MPEG2-PS decoding section 111 starts playing back PS #2. Thereafter, 5 the reading section 121 reproduces the overlapping audio frame while playing back PS3. In this case, the MPEG2-PS decoding section 111 delays its playback timing by the amount of overlap and then starts playing it. However, if the playback timing were delayed at every connection point, then the video- 10 audio gap might broaden to a sensible degree. That is why it is necessary to read and output the audio frame of PS #3 at its original playback timing without using OVRP0001.AC3 all through the playback range.

The mute interval caused by the audio gap can be 15 eliminated by any of these playback processes. In any of the examples shown in portions (a) and (b) of FIG. 40, only some of the audio samples in the overlapping PS track (i.e., only the audio data corresponding to the overlap range) may be discarded and the remaining audio data may be played back at 20 playback timings originally specified by PTS. Even so, the

mute interval caused by the audio gap can also be eliminated during the playback.

FIG. 41 shows an example in which the playback ranges PS #1 and PS #3 are connected together so as to be played back seamlessly using a play list without directly editing them. In FIG. 39, a moving picture file is actually edited by connecting the moving picture streams PS #1 and PS #3 together. In FIG. 41 on the other hand, their relationship is just described using a play list file. One audio frame including an overlapping portion is written just before MOVE003.MPG. The play list MOVE0001.PLF includes a PS track for PS #1, an audio track for an audio frame including the overlapping portion, and a PS track for PS #3, and describes Edit List Atoms of the respective tracks so as to realize the playback timings shown in FIG. 40.

It should be noted that if two moving picture streams are connected together using the play list shown in FIG. 41, then video streams in the moving picture streams usually do not satisfy the VBV buffer conditions of the MPEG-2 Video standard before and after the connection point unless subjected to an

editing process. Accordingly, in connecting video seamlessly, the reading control section and MPEG-2 decoding section need to play back seamlessly streams that do not satisfy the VBV buffer conditions.

5 FIG. 42 shows the data structure of Sample Description Entry of the play list. Seamless information includes fields for a seamless flag, audio discontinuity information, SCR discontinuity information, an STC continuity flag, and audio control information. If the seamless flag is zero in Sample
10 Description Entry of the play list, then there is no need to set any values for the recording start date, presentation start time, presentation end time and discontinuity start flag. On the other hand, if the seamless flag is one, then appropriate values need to be set as in the auxiliary
15 information file for initial recording. This is because in a play list, Sample Description Entry needs to be used in common by a plurality of chunks and these fields cannot always be effective in that case.

FIG. 43 shows the data structure of the seamless
20 information. In the fields shown in FIG. 43, each field

having the same name as the counterpart shown in FIG. 19 has the same structure as it. STC continuity information=1 shows that a system time clock (of 27 MHz), used as a reference for the previous stream, is continuous with an STC value that is used as a reference by this stream. More specifically, it shows that a PTS, a DTS, and an SCR are applied to a moving picture file based on the same STC value and are continuous with each other. The audio control information shows whether or not the audio at the PS connection point needs to be once faded out and then faded in. By reference to this field, the player controls the fade-out of the audio just before the connection point and the fade-in of the audio right after the connection point just as described in the play list. In this manner, the audio can be controlled appropriately according to the contents of the audio before and after the connection point. For example, if the audio frequency characteristic after the connection point is quite different from that before the connection point, then the audio is preferably faded out once and then faded in. On the other hand, if those frequency characteristics are similar to each other, then neither fade-

out nor fade-out is preferred.

FIG. 44 shows the values of the seamless flag and STC continuity information in Sample Description Entry in a situation where two moving picture files MOVE0001.MPG and
5 MOVE0003.MPG are connected seamlessly together with a bridge file MOVE0002.MPG interposed between them by describing a play list including the bridge file.

The bridge file is a moving picture file MOVE0002.MPG including a connecting portion of PS #1 and PS #3. The video
10 streams in the two moving picture streams are supposed to satisfy the VBV buffer conditions of the MPEG-2 Video standard before and after this connecting portion. That is to say, the data structure is supposed to be the same as that shown in FIG. 39.

15 Each of these moving picture files has a predetermined duration (of 10 seconds to 20 seconds, for example) as in FIG. 37. A post recording data area is provided physically just before the moving picture stream of the predetermined duration. And post recording empty areas, which are unused
20 areas, are reserved as separate files named MOVE0001.EMP,

MOVE0002.EMP and MOVE0003.EMP.

FIG. 45 shows the data structure of Edit List Atom of the play list shown in FIG. 44. This play list includes a PS track for an MPEG-2 PS and an audio track for AC-3 audio. The PS track makes reference to MOVE0001.MPG, MOVE0002.MPG and MOVE0003.MPG shown in FIG. 44 by way of Data Reference Atom. Meanwhile, the audio track makes reference to OVRP0001.AC3 file, including one audio frame, by way of Data Reference Atom, too. In Edit List Atom of the PS track, Edit List Table representing four playback ranges is stored. The respective playback ranges #1 through #4 correspond to the playback ranges #1 through #4 shown in FIG. 44, respectively. On the other hand, in Edit List Atom of the audio frame stored in the post recording area, Edit List Table, representing pause interval #1, playback range and pause interval #2, is stored. If the reading section reads this play list, the audio track is supposed to be read preferentially without reading the audio from the PS track in a range where the playback of the audio track is specified. As a result, the audio frame stored in the post recording area is played back in the audio gap

interval. And when that audio frame has been played back, the audio frame in the overlapping PS #3 and following audio frames will be played back after having been delayed by the amount of overlap. Alternatively, after the audio frame in PS #3, including the audio data to play back immediately after that, has been decoded, only the non-overlapping remaining portion is played back.

In Edit List Table, track_duration specifies the video duration of the playback range and media_time specifies the location of the playback range in the moving picture file. As the location of this playback range, the location of the top video of the playback range is represented as a time offset value, which is defined by regarding the top of the moving picture file as time zero. Media_time = -1 means a pause interval in which nothing is played back during track_duration. As media_rate, 1.0, meaning 1x playback, is set. The reading section reads Edit List Atom from both the PS track and the audio track alike, thereby carrying out a playback control based on it.

FIG. 46 shows the data structure of Sample Description

Atom in the audio track shown in FIG. 45 (where the audio data is supposed to comply with Dolby AC-3 format). Sample_description_entry includes audio seamless information. This audio seamless information includes an overlap location, 5 which shows whether the audio overlap is supposed to be done at the top of one audio frame or at the end thereof. The audio seamless information also includes an overlap period as time information, which is counted in response to a clock pulse of 27 MHz. By reference to this overlap location and 10 overlap period, the playback of audio is controlled around the overlapping range.

In this manner, a play list for realizing seamless playback of video and audio can be provided such that its form is compatible with a conventional stream that is supposed to 15 have an audio gap. That is to say, either a seamless playback using the audio gap or a seamless playback using the overlapping audio frame may be selected arbitrarily. Consequently, even an apparatus that cope with only the conventional audio gap can perform the seamless playback at 20 the stream connection point at least by the conventional

method.

In addition, the connection point can be controlled finely according to the contents of the audio.

Besides, Sample Description Entry, which cuts down the
5 redundancy of an MP4 file play list and which can provide detailed description required for a seamless play list, is realized.

According to the present invention, the seamless playback of video and audio is realized by recording the overlapping
10 portion of the audio. Alternatively, the video and audio may be played back pseudo-seamlessly by skipping the playback of a video frame without using that overlapping portion.

Also, in the preferred embodiment described above, the overlapping portion of the audio is recorded in the post
15 recording area. Alternatively, the overlapping portion may also be stored in Movie Data Atom in the play list file. In AC3, for example, a single frame may have a data size of several kilobytes. Optionally, instead of the STC continuity flag shown in FIG. 43, the presentation end time of the PS
20 just before the connection point and the presentation start

time of the PS right after the connection point may be recorded. In that case, if the seamless flag is one and if the presentation end and start times are equal to each other, then it may mean that STC continuity flag is one. As another
5 alternative, instead of the STC continuity flag, the difference between the presentation end time of the PS just before the connection point and the presentation start time of the PS right after the connection point may also be recorded. In that case, if the seamless flag is one and if the
10 difference between the presentation end and start times is zero, then it may mean that STC continuity flag is one.

In the present invention, only the audio frame including the audio overlapping portion is recorded in the post recording area separately from the portion of PS #3.
15 Alternatively, both the extended portion shown in FIG. 40 and the portion of the audio frame, including the overlapping portion shown in portion (a) or (b) of FIG. 40, may be recorded in the post recording area. Furthermore, an audio frame, associated with the top video of PS #3, may also be
20 recorded continuously in the post recording area. In that

case, the audio switching interval will become longer between the audio in the PS track and the audio in the audio track. As a result, seamless playback is realized even more easily by using the audio overlap technique. In those cases, the audio switching interval may be controlled with Edit List Atom of the play list.

The audio control information is included in the seamless information of a PS track. Optionally, the audio control information may also be included in the seamless information of an audio track as well. Even so, the fade-out and fade-in are also controlled in a similar manner just before and right after the connection point.

A method of playing back an audio frame continuously before and after a connection point without applying fade-out or fade-in to the connection point was mentioned. This is a technique effectively applicable to AC-3, MPEG Audio Layer 2 and other compression methods.

In the preferred embodiments of the present invention described above, the MPEG2-PS 14 shown in FIG. 12 is supposed to contain moving picture data (VOBU) for 0.4 second to 1

second. However, the time range may be different. Also, the MPEG2-PS 14 is supposed to consist of VOBUs compliant with the DVD Video recording standard. Alternatively, the MPEG2-PS 14 may also be a program stream compliant with the MPEG-2 System standard or a program stream compliant with the DVD Video standard.

In the preferred embodiments of the present invention described above, the overlapping audio is supposed to be recorded in the post recording area. Alternatively, the overlapping audio may also be recorded elsewhere but its storage location is preferably as physically close to the moving picture file as possible.

The audio file is supposed to be made up of AC-3 audio frames. Optionally, the audio file may also be stored in either an MPEG-2 program stream or in an MPEG-2 transport stream.

In the data processor 10 shown in FIG. 11, the storage medium 131 is supposed to be a DVD-RAM disk. However, the storage medium is not particularly limited to a DVD-RAM. Examples of other preferred storage media 131 include optical

storage media such as an MO, a DVD-R, a DVD-RW, a DVD+RW, a Blu-ray, a CD-R and a CD-RW and magnetic recording media such as a hard disk. As another alternative, the storage medium 131 may also be a semiconductor storage medium including a semiconductor memory such as a flash memory card. Optionally, the storage medium may even use a hologram. Furthermore, the storage medium may be either removable from, or built in, the data processor.

The data processor 10 performs the processing of generating, writing and reading a data stream according to a computer program. For example, the processing of generating and writing the data stream may be carried out by executing a computer program that is described based on the flowchart shown in FIG. 21. The computer program may be stored in any of various types of storage media. Examples of preferred storage media include optical storage media such as optical disks, semiconductor storage media such as an SD memory card and an EEPROM, and magnetic recording media such as a flexible disk. Instead of using such a storage medium, the computer program may also be downloaded via a

telecommunications line (e.g., through the Internet, for example) and installed in the optical disc drive 100.

The file system is supposed to be compliant with UDF but may also be compliant with FAT, NTFS or any other standard.

5 The video is supposed to be an MPEG-2 video stream but may also be an MPEG-4 AVC, for example. Also, the audio is supposed to be compliant with AC-3 but may also be compliant with LPCM, MPEG-Audio or any other appropriate standard. Furthermore, the moving picture stream is supposed to have a
10 data structure of an MPEG-2 program stream, for example, but may also be any other type of data stream as long as the video and audio are multiplexed together.

INDUSTRIAL APPLICABILITY

15 According to the present invention, while the data structure of auxiliary information is adapted to the up-to-date standard so as to comply with the ISO standard, a data structure for a data stream, having a format equivalent to a conventional one, and a data processor, operating on such a
20 data structure, are provided as well. Since the data stream

is compatible with the conventional format, any existent application can use the data stream. Consequently, every piece of existent software and hardware can be used effectively. In addition, the present invention also
5 provides a data processor that can play back not just video but also audio without a break at all when two moving picture streams are combined together by editing. Furthermore, since the data processor is still compatible with the conventional data stream, compatibility with existent playback equipment
10 is guaranteed, too.